



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/809,152

03/24/2004

Patrick L. Edson

MWS-104RCE2

7394

74321

7590

03/15/2010

LAHIVE & COCKFIELD, LLP/THE MATHWORKS
FLOOR 30, SUITE 3000
One Post Office Square
Boston, MA 02109-2127

EXAMINER

ALVESTEFFER, STEPHEN D

ART UNIT

PAPER NUMBER

2175

MAIL DATE

DELIVERY MODE

03/15/2010

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/809,152	Applicant(s) EDSON ET AL.	
	Examiner Stephen Alvesteffer	Art Unit 2175	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 15 December 2009.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-10, 12-36, 38-50 and 52-59 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-10, 12-36, 38-50 and 52-59 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Response to Amendment

This Office Action is responsive to the Amendment filed December 15, 2009. Claims 30 and 50 are amended. Claims 2, 11, 37, and 51 are previously cancelled. Claims 1, 30, 44, and 59 are independent. Claims 1, 2-10, 12-36, 38-50, and 52-59 remain pending.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1, 3-6, 25, 28, 56, and 59 are rejected under 35 U.S.C. 102(b) as being anticipated by Johnson et al. (hereinafter Johnson), United States Patent Application Publication number 2003/0001896.

Regarding claim 1, Johnson teaches a computer readable storage medium storing computer executable instructions that when executed on a processor manage a graphical interface, the medium storing:

instructions for providing a graphical interface, a hardware device and a software device being accessible through the graphical interface, the software device being accessible to a computer (see paragraph [0125]; *"a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a*

Art Unit: 2175

measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a plurality of measurement devices. In one embodiment, at least one of the plurality of measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device”);

instructions for providing at least one interactive hardware object accessible to the computer, where the hardware object represents the hardware device and is depicted in the graphical interface, the hardware object interacting with the hardware device (see Johnson paragraph [0099]; *“the graphical icon that visually represents the node represents the function, and the underlying program instructions and/or data structures which are represented by the node graphical icon are actually performing the function. Thus the specification and claims of the present application refer generally to a node performing a function, it being understood that the node includes or represents underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element) to perform the function”);*

instructions for providing a software object, wherein the software object is representative of the software device, where the software object is depicted in the graphical interface and is configured to be interactive with the software device (see paragraph [0255]; *“Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the*

Art Unit: 2175

measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12”; see also paragraph [0107]; “The run-time builder may also provide various parameters to hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time 790. In other words, the run-time builder 780 may configure one or more measurement devices according to the run-time specification 770”);

instructions for receiving, from a user, a plurality of configurations of the hardware device, each configuration allowing the user to edit at least one property of the hardware object (see Figure 26 and paragraph [0138]; *“the GUI may display a third panel, e.g., a channel configuration panel, which presents options for specifying values of one or more parameters for the indicated channel(s)”*);

instructions for displaying the plurality of configurations simultaneously, wherein each configuration corresponds to a unique hardware object that represents the hardware device (see Johnson Figure 26 and paragraph [0138]; *“the GUI may display a third panel, e.g., a channel configuration panel, which presents options for specifying values of one or more parameters for the indicated channel(s)”*; see also Johnson paragraph [0012]; *“The GUI may receive user input characterizing the measurement task, where the user input indicates values for a plurality of parameters of the measurement task. For example, the parameters may include five or more of measurement type, device type, channel parameters, sampling parameters, trigger*

Art Unit: 2175

parameters, clock parameters, scaling parameters, synchronization parameters, routing parameters, and data publishing parameters”);

instructions for receiving, from a user, a selection of one configuration from the plurality of configurations (for example, Johnson Figure 18A, showing two configurations, Min and Max, allowing the user to select one of the configurations for editing); and

instructions for communicating with the hardware device corresponding to the selected configuration using the selected configuration (for example, Johnson Figure 18A, a user selecting and editing either the Min or Max configurations).

Regarding claim 3, Johnson teaches providing an analysis object, wherein said analysis object is adapted to communicate with at least one of said hardware object and said software object for analysis of data from at least one of said hardware object and said software object (see paragraph [0255]; “*Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12*”).

Regarding claim 4, Johnson teaches instructions for receiving code for execution by the hardware object (see Johnson paragraph [0099]; “*underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element...*”).

Regarding claim 5, Johnson teaches that a plurality of hardware objects are provided for a single hardware device (see paragraph [0100]; Johnson's invention allows several different types of nodes to be created to accomplish various measurement tasks such as reading and writing to and from a measurement device).

Regarding claim 6, Johnson teaches that a plurality of hardware objects are provided for a plurality of hardware devices. (see paragraph [0100]; Johnson's invention allows several different types of nodes to be created to accomplish various measurement tasks such as reading and writing to and from a measurement device).

Regarding claim 25, Johnson teaches that the graphical interface is implemented with an extensible API (see Johnson paragraph [0158]).

Regarding claim 28, Johnson teaches that the graphical interface is adapted to operate on a plurality of operating systems (see Johnson paragraph [0053]; Although Johnson does not specify exactly which operating systems or exactly how many operating systems his invention supports, it is inherent and well-known in the art that software code is capable of executing on more than one different operating system).

Regarding claim 56, Johnson teaches that the hardware object enables communication between the graphical interface and the hardware device, and the software object enables communication between the graphical interface and the software device (see paragraph [0125]; *"a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a*

plurality of measurement devices. In one embodiment, at least one of the plurality of measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device”).

Regarding claim 59, Johnson teaches a computer readable storage medium storing computer executable instructions that when executed on a processor manage a graphical interface, the medium storing:

instructions for providing a graphical interface, at least one hardware device and one software device being accessible through the graphical interface (see paragraph [0125]; *“a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a plurality of measurement devices. In one embodiment, at least one of the plurality of measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device”)*), the graphical interface being updated in response to a change in the hardware device or the software device (see Johnson Figure 26 and paragraph [0242]; *“parameters specific to the selected device may be configured in this panel. In this example, logic parameters related to the PCI-MIO-16E-1 device are shown, including high/low state levels, idle line state, and idle state pattern. Of course, when other devices are selected, other corresponding parameters and controls may be presented”)*);

instructions for providing a plurality of hardware objects accessible to the computer, where each of the hardware objects represents a hardware device and is depicted in the graphical interface, each hardware object configured to be interactive with the hardware device (see paragraph [0099]; *“the graphical icon that visually represents the node represents the function, and the underlying program instructions and/or data structures which are represented by the node graphical icon are actually performing the function. Thus the specification and claims of the present application refer generally to a node performing a function, it being understood that the node includes or represents underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element) to perform the function”*);

instructions for providing a plurality of software objects, each representative of a software device accessible to the computer, where each of the software objects is depicted in the graphical interface and is configured to be interactive with the software device (see paragraph [0255]; *“Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12”*; see also paragraph [0107]; *“The run-time builder may also provide various parameters to hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time*

Art Unit: 2175

790. *In other words, the run-time builder 780 may configure one or more measurement devices according to the run-time specification 770”);*

instructions for providing a plurality of configurations of the hardware object, each configuration allowing the user to edit at least one property of the hardware object (see Figure 26 and paragraph [0138]; *“the GUI may display a third panel, e.g., a channel configuration panel, which presents options for specifying values of one or more parameters for the indicated channel(s)”);*

instructions for displaying the plurality of hardware objects and the plurality of software objects and at least one of the plurality of configurations of one of the hardware objects or one of the software objects to a user in a single graphical interface simultaneously (see Johnson Figure 16, Figure 29 showing “Voltage” drop-down, and paragraph [0136]; *“the GUI may display a second panel, e.g., a channels selection panel, which presents a list of available devices and corresponding channels. The available devices may correspond to the indicated measurement type. For example, if the selected measurement type were voltage, the devices listed may be those devices available to the system which are suitable for measurement a voltage. An example of the device and channel list is shown in FIG. 16”);*

instructions for receiving, from a user, a selection of one configuration from the plurality of configurations (for example, Johnson Figure 18A, showing two configurations, Min and Max, allowing the user to select one of the configurations for editing); and

instructions for communicating with the hardware device corresponding to the selected configuration using the selected configuration (for example, Johnson Figure 18A, a user selecting and editing either the Min or Max configurations).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 7, 8, and 12-14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra* and Fuller, III et al. (hereinafter Fuller), United States Patent Application Publication number 2003/0035008.

Regarding claim 7, Johnson teaches every limitation of claim 7 except instructions for scanning for available hardware; and instructions for creating an additional hardware object for each hardware device detected and not already associated with a hardware object. Fuller teaches a method and apparatus for controlling an instrumentation system that automatically scans for available hardware (instruments) and allowing users to select hardware (instruments) from a list of detected hardware (instruments) (see paragraph [0020], "*the computer system may automatically detect the one or more message-based instruments that are connected to the computer system. In other words, the computer system may automatically scan for message-based instruments coupled to the system*"). It would have been obvious to a person of

Art Unit: 2175

ordinary skill in the art at the time the invention was made to combine the scanning for available hardware of Fuller with the invention of Johnson in order to allow custom hardware components to be added to the system.

Regarding claim 8, Johnson/Fuller teaches all the steps of claim 8 except that instructions for scanning involves instructions for receiving user-defined commands to be sent to the hardware device to attempt to identify the hardware device. Fuller teaches allowing the user to initiate a hardware scan. A user-initiated hardware scan is being interpreted with the broadest reasonable interpretation to be the same as sending user-defined command to a hardware device (see paragraph [0020], “*A user interface (UI) may be provided that allows the user to initiate a scan for message-based instruments. The user may scroll through and select an instrument from a list of detected instruments, or may otherwise specify a particular instrument to be communicated with*”). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the user-initiated hardware scan of Fuller with the invention of Johnson in order to allow custom hardware components to be added to the system on demand.

Regarding claim 12, Johnson/Fuller teaches every limitation of claim 12 except that at least one of instructions for providing at least one hardware object and providing at least one software object further comprises instructions for accessing at least one of a hardware object and a software object located on a remote computer. Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see paragraph [0168]; “*Tasks may be collected and organized for*

Art Unit: 2175

distribution, for example through a website"). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of Fuller with the invention of Johnson in order to allow measurement or testing over a network.

Regarding claim 13, Johnson/Fuller teaches every limitation of claim 13 except that instructions for accessing is performed through a web page. Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see paragraph [0168]; "*Tasks may be collected and organized for distribution, for example through a website*"). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of Fuller with the invention of Johnson in order to allow measurement or testing over a network.

Regarding claim 14, Johnson/Fuller teaches every limitation of claim 14 except that instructions for accessing is performed over a network. Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see paragraph [0168]; "*Tasks may be collected and organized for distribution, for example through a website*"). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of Fuller with the invention of Johnson in order to allow measurement or testing over a network.

Claims 9 and 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra* and Hsiung et al. (hereinafter Hsiung), United States Patent Application Publication number 2003/0083756.

Regarding claim 9, Johnson teaches all the elements of claim 9 except that the analysis object filters data. Hsiung teaches a system for monitoring industrial components with an analysis component that performs filtering (see paragraph [0056]; *“The upload process takes data from the acquisition device and uploads them into the main process manager 314 for processing. Here, the data are in electronic form. In embodiments where the data has been stored in data storage, they are retrieved and then loaded into the process. Preferably, the data can be loaded onto workspace to a text file or loaded into a spread sheet for analysis. Next, the filter process 302 filters the data to remove any imperfections”*). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the filtering of data of Hsiung with the invention of Johnson for the purpose of providing data analysis functionality.

Regarding claim 10, Johnson teaches all the elements of claim 10 except that the analysis object plots data. Hsiung teaches a system for monitoring industrial components with an analysis component that performs plotting of data (see paragraph [0058]; *“A baseline correction process may also find response peaks, calculate $\Delta R/R$, and plot the $\Delta R/R$ verses time stamps, where the data have been captured”*). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the plotting of data of Hsiung with the invention of Johnson for the purpose of providing data analysis functionality.

Claim 15 is rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra*, Fuller (2003/0035008) *supra*, and Hsiung (2003/0083756) *supra*.

Regarding claim 15, Johnson/Fuller teach every limitation of claim 15 except that instructions for accessing is performed by passing commands over the network in a MATLAB environment. Hsiung teaches using MATLAB in association with the invention (see paragraph [0534]; “*Multi-way PCA is a natural choice since PCA is already included, algorithms are available for evaluation in Matlab toolboxes, and the technique serves as a good benchmark when discussing benefits of other algorithms*”). It would have been obvious to one of ordinary skill in the art that the MATLAB environment could be used as taught by Hsiung with the invention taught by Johnson/Fuller.

Claims 16, 17, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra* and Schmit et al. (hereinafter Schmit), United States Patent Application number 2003/0004670.

Regarding claim 16, Johnson teaches every limitation of claim 16 except instructions for modifying at least one of the hardware object and the software object. Schmit teaches a system and method for building a measurement system in which the most efficient protocol to use with each measurement device is determined and applied (see Schmit paragraph [0500]). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the hardware protocol selection

system of Schmit with the invention of Johnson for the purpose of making the measurement system more efficient.

Regarding claim 17, Johnson/Schmit teaches that modifying specifies a protocol for use by the hardware object for communication with the hardware device (see Schmit paragraph [0500]).

Regarding claim 27, Johnson/Schmit teaches instructions for generating an analysis object that can be used in SIMULINK (see Johnson paragraph [0101]; Johnson's invention makes use of the LabVIEW environment for generating analysis objects; see also Schmit paragraph [0619]; Schmit teaches that SIMULINK is similar in function to LabVIEW).

Claims 18-24, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra*, Hsiung (2003/0083756) *supra*, and Pike et al. (hereinafter Pike), United States Patent Application Publication number 2003/0056018.

Regarding claim 18, Johnson/Hsiung teaches every limitation of claim 18 except that modifying modifies a value stored in an array of an array-based environment. Pike teaches a system for linking users to control instruments wherein an array-based environment can be used to change the properties of the control instruments (see Pike paragraph [0010]; “*The user may also create an object array in response to an array creation command. The object array includes as elements, a first and a second instrument object. The user may change the properties of the first and second communication channels by changing properties of the object array*”; see also Pike

Art Unit: 2175

paragraph [0070]; *"The array-based environment 104 includes functions used by the user 30 to create an instrument object 108 through function calls 46, as well as to configure an instrument object's properties and to connect the instrument object with one of the control instruments 22"*). Pike further teaches that the graphical user interface can be used to export data to an array-based environment such as MATLAB (see paragraph [0040]; *"User 30 may send a list of requests or commands to processor 20 from the GUI 14 to establish a communication channel between the computer 12 and the control instruments 22. The user 30 does so by writing a user program 80, which resides in memory 26 of computer 12. The user program 80 may be associated with the syntax of, for example, any interpreted programming environment. An interpreted programming environment may be any proprietary program that performs mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments, among many others. An example of an interpreted programming environment is MATLAB.RTM. from MathWorks, Inc., of Natick, Mass"*). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the array-based environment steps of Pike with the measurement systems of Johnson/Hsiung in order to provide array-based control of the measurement devices.

Regarding claim 19, Johnson/Hsiung/Pike teaches instructions for modifying a value stored in an array of an array-based environment, thereby modifying at least one of the hardware object and the software object (see Pike paragraphs [0010], [0070], and [0040]).

Regarding claim 20, Johnson/Hsiung/Pike teaches instructions for exporting data from the graphical interface to an array-based environment (see Pike paragraphs [0010], [0070], and [0040]).

Regarding claim 21, Johnson/Hsiung/Pike teaches instructions for converting user actions with the graphical interface into code (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted programming code capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 22, Johnson/Hsiung/Pike teaches that the code is created in a MATLAB environment (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted programming code capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 23, Johnson/Hsiung/Pike teaches that the code comprises steps to create an analysis object, configure the analysis object and write and read data from the analysis object (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted programming code capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 24, Johnson/Hsiung/Pike teaches that the code comprises an analysis routine (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted programming code

Art Unit: 2175

capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 26, Johnson/Hsiung/Pike teaches instructions for generating an analysis object so that the analysis object can be used in MATLAB (see Pike paragraph [0040]).

Claim 29 is rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra* and Phathayakorn et al. (hereinafter Phathayakorn), United States Patent number 5,986,653.

Regarding claim 29, Johnson teaches every limitation of claim 29 except that the graphical interface comprises a tree view, wherein the tree view groups the hardware objects and the software objects by a functionality characteristic. Tree views of hardware and software objects grouped by functionality were a well-known graphical user interface technique at the time the invention was made. Phathayakorn shows selecting a functional group of objects from a tree view graphical representation (see Figures 2A-5B). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the tree view graphical representation of Phathayakorn to the invention of Johnson in order to provide a representation of the devices on the user interface.

Claims 30-33, 36, 44-47, 50, 57, and 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra* and Gray et al. (hereinafter Gray), United States Patent 6,185,491.

Regarding claim 30, Johnson substantially teaches a method for managing an interface, the method comprising:

providing a graphical interface that provides interaction with an array-based environment, a hardware device and a software device being accessible through the graphical interface, the software device being accessible to a computer (see paragraph [0125]; *“a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a plurality of measurement devices. In one embodiment, at least one of the plurality of measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device”*);

providing at least one hardware object accessible to the computer, where the hardware object represents the hardware device and is depicted in the graphical interface, the hardware object configured to be interactive with the hardware device (see paragraph [0099]; *“the graphical icon that visually represents the node represents the function, and the underlying program instructions and/or data structures which are represented by the node graphical icon are actually performing the function. Thus the specification and claims of the present application refer generally to a node performing*

Art Unit: 2175

a function, it being understood that the node includes or represents underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element) to perform the function”);

providing at least one software object, representative of the software device, where the software object is depicted in the graphical interface, and is configured to be interactive with the software device (see paragraph [0107]; *“The run-time builder may also provide various parameters to hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time 790. In other words, the run-time builder 780 may configure one or more measurement devices according to the run-time specification 770”*) and;

updating the graphical interface when the hardware object or the software object are changed in the array-based environment (see Gray, addressed below); and

displaying the hardware object and the software object to a user (see Johnson Figure 16 and paragraph [0136; *“the GUI may display a second panel, e.g., a channels selection panel, which presents a list of available devices and corresponding channels. The available devices may correspond to the indicated measurement type. For example, if the selected measurement type were voltage, the devices listed may be those devices available to the system which are suitable for measurement a voltage. An example of the device and channel list is shown in FIG. 16”*).

Johnson does not disclose updating the graphical interface when the hardware object or the software object are changed in the array-based environment. Gray

Art Unit: 2175

teaches a vehicle having networked components connected in an array in which if one component is removed, the graphical interface will be updated to reflect that the component is no longer connected to the array (see Gray column 7 lines 36-46; "*When a device is detected as having been removed, the device interface, previously received from the device is removed from memory (1420) and the data structure is updated to remove the device entries (1430).*"). It would have been obvious to one having ordinary skill in the art at the time the invention was made to allow users to physically remove components from the system and to update the graphical interface to reflect the changes as taught by Gray in the invention of Johnson so that it would be intuitive to add or remove components to the system.

Regarding claim 31, Johnson/Gray teaches receiving code for execution by the hardware object (see paragraph [0255]; "*Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12*").

Regarding claim 32, Johnson/Gray teaches that at least one additional hardware object is provided for the hardware device (see paragraph [0100]; Johnson's invention allows several different types of nodes to be created to accomplish various measurement tasks such as reading and writing to and from a measurement device).

Regarding claim 33, Johnson/Gray teaches that additional hardware objects are provided for a plurality of hardware devices (see paragraph [0100]; Johnson's invention

Art Unit: 2175

allows several different types of nodes to be created to accomplish various measurement tasks such as reading and writing to and from a measurement device).

Regarding claim 36, Johnson/Gray teaches providing an analysis object adapted to communicate with at least one of the hardware object and the software object (see Johnson paragraph [0099]; *“underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element...)”*).

Regarding claim 44, Johnson/Gray teaches a computing device comprising:
an array-based environment (see Gray column 7 lines 36-46; *“When a device is removed from its bus connection (1400), the vehicle control center detects that a device previously installed is no longer connected (1410)”*, the bus connection is the same as an array-based environment);

a storage medium for storing and a processor for processing (see Johnson claim 41; *“a processor; and a memory medium...”*);

a graphical interface, at least one hardware device and one software device being accessible through the graphical interface (see paragraph [0125]; *“a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a plurality of measurement devices. In one embodiment, at least one of the plurality of measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device”*);

a plurality of hardware objects accessible to the computer, where each of the hardware objects represents a hardware device and is depicted in the graphical interface, each hardware object configured to be interactive with the hardware device (see paragraph [0099]; *“the graphical icon that visually represents the node represents the function, and the underlying program instructions and/or data structures which are represented by the node graphical icon are actually performing the function. Thus the specification and claims of the present application refer generally to a node performing a function, it being understood that the node includes or represents underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element) to perform the function”*);

a plurality of software objects, each representative of a software device accessible to the computer, where each of the software objects is depicted in the graphical interface and is configured to be interactive with the software device (see paragraph [0255]; *“Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12”*; see also paragraph [0107]; *“The run-time builder may also provide various parameters to hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time 790. In*

Art Unit: 2175

other words, the run-time builder 780 may configure one or more measurement devices according to the run-time specification 770"); and

a display device to display the plurality of hardware objects and the plurality of software objects and at least one configuration of one of the hardware objects or one of the software objects to a user in a single graphical interface simultaneously (see Johnson Figure 16, Figure 29 showing "Voltage" drop-down box, and paragraph [0136; "the GUI may display a second panel, e.g., a channels selection panel, which presents a list of available devices and corresponding channels. The available devices may correspond to the indicated measurement type. For example, if the selected measurement type were voltage, the devices listed may be those devices available to the system which are suitable for measurement a voltage. An example of the device and channel list is shown in FIG. 16"), wherein the plurality of hardware objects and the plurality of software objects are accessible through both the array-based environment and the graphical interface (see Gray column 7 lines 36-46; "When a device is detected as having been removed, the device interface, previously received from the device is removed from memory (1420) and the data structure is updated to remove the device entries (1430).").

Regarding claim 45, Johnson/Gray teaches that the system receives code for execution by the hardware objects (see paragraph [0255]; *"Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a*

run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12”).

Regarding claim 46, Johnson/Gray teaches that a plurality of hardware objects are provided for a single hardware device (see paragraph [0100]; Johnson’s invention allows several different types of nodes to be created to accomplish various measurement tasks such as reading and writing to and from a measurement device).

Regarding claim 47, Johnson/Gray teaches that a plurality of hardware objects are provided for a plurality of hardware devices (see paragraph [0100]; Johnson’s invention allows several different types of nodes to be created to accomplish various measurement tasks such as reading and writing to and from a measurement device).

Regarding claim 50, Johnson/Gray teaches that an analysis object is provided adapted to communicate with at least one of the hardware objects and the software objects (see Johnson paragraph [0099]; “*underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element...)*”).

Regarding claim 57, Johnson/Gray teaches that the hardware object enables communication between the graphical interface and the hardware device, and the software object enables communication between the graphical interface and the software device (see paragraph [0125]; “*a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a plurality of measurement devices. In one embodiment, at least one of the plurality of*

Art Unit: 2175

measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device”).

Claim 58 recites a computing device having substantially the same limitations as the method of claim 57. Therefore, claim 58 is rejected under the same rationale.

Claims 34, 35, 38, 48, 49, and 52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra*, Gray (6,185,491) *supra*, and Fuller (2003/0035008) *supra*.

Regarding claim 34, Johnson/Gray teaches every limitation of claim 34 except scanning for available hardware; and creating an additional hardware object for each hardware device detected and not already associated with a hardware object. Fuller teaches a method and apparatus for controlling an instrumentation system that automatically scans for available hardware (instruments) and allowing users to select hardware (instruments) from a list of detected hardware (instruments) (see paragraph [0020], “*the computer system may automatically detect the one or more message-based instruments that are connected to the computer system. In other words, the computer system may automatically scan for message-based instruments coupled to the system*”). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the scanning for available hardware of Fuller with the invention of Johnson/Gray in order to allow custom hardware components to be added to the system.

Regarding claim 35, Johnson/Gray teaches all the steps of claim 35 except that scanning involves instructions for receiving user-defined commands to be sent to the hardware device to attempt to identify the hardware device. Fuller teaches allowing the user to initiate a hardware scan. A user-initiated hardware scan is being interpreted with the broadest reasonable interpretation to be the same as sending user-defined command to a hardware device (see Fuller paragraph [0020], “*A user interface (UI) may be provided that allows the user to initiate a scan for message-based instruments. The user may scroll through and select an instrument from a list of detected instruments, or may otherwise specify a particular instrument to be communicated with*”). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the user-initiated hardware scan of Fuller with the invention of Johnson/Gray in order to allow custom hardware components to be added to the system on demand.

Regarding claim 38, Johnson/Gray teaches every limitation of claim 38 except that at least one of providing at least one hardware object and providing at least one software object further comprises accessing at least one of a hardware object and a software object located on a remote computer. Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see paragraph [0168]; “*Tasks may be collected and organized for distribution, for example through a website*”). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of Fuller with the invention of Johnson/Gray in order to allow measurement or testing over a network.

Claims 48, 49, and 52 recite a system with substantially the same limitations as claims 34, 35, and 38, respectively. Therefore, the claims are rejected under the same rationale.

Claims 39, 40, 43, 53, and 54 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra*, Gray (US 6,185,491) *supra*, and Schmit (US 2003/0004670) *supra*.

Regarding claim 39, Johnson/Gray teaches every limitation of claim 39 except modifying at least one of the hardware object and the software object. Schmit teaches a system and method for building a measurement system in which the most efficient protocol to use with each measurement device is determined and applied (see Schmit paragraph [0500]). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the hardware protocol selection system of Schmit with the invention of Johnson/Gray for the purpose of making the measurement system more efficient.

Regarding claim 40, Johnson/Gray/Schmit teaches that modifying specifies a protocol for use by the hardware object for communication with the hardware device (see Schmit paragraph [0500]).

Regarding claim 43, Johnson/Gray/Schmidt teaches generating an analysis object that can be used in SIMULINK (see Schmit paragraph [0619]).

Claims 53 and 54 recite a system with substantially the same limitations as claims 39 and 40. Therefore, claims 53 and 54 are rejected under the same rationale.

Claims 41, 42, and 55 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (2003/0001896) *supra*, Gray (US 6,185,491) *supra*, and Pike (US 2003/0056018) *supra*.

Regarding claim 41, Johnson/Gray teaches every limitation of claim 41 except that modifying modifies a value stored in an array of an array-based environment. Pike teaches a system for linking users to control instruments wherein an array-based environment can be used to change the properties of the control instruments (see Pike paragraph [0010]; “*The user may also create an object array in response to an array creation command. The object array includes as elements, a first and a second instrument object. The user may change the properties of the first and second communication channels by changing properties of the object array*”; see also Pike paragraph [0070]; “*The array-based environment 104 includes functions used by the user 30 to create an instrument object 108 through function calls 46, as well as to configure an instrument object's properties and to connect the instrument object with one of the control instruments 22*”). Pike further teaches that the graphical user interface can be used to export data to an array-based environment such as MATLAB (see paragraph [0040]; “*User 30 may send a list of requests or commands to processor 20 from the GUI 14 to establish a communication channel between the computer 12 and the control instruments 22. The user 30 does so by writing a user program 80, which resides in memory 26 of computer 12. The user program 80 may be associated with the syntax of, for example, any interpreted programming environment. An interpreted programming environment may be any proprietary program that performs mathematical*

Art Unit: 2175

computations for modeling, simulation, graphics, or data analysis related to control instruments, among many others. An example of an interpreted programming environment is MATLAB.RTM. from MathWorks, Inc., of Natick, Mass”). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the array-based environment steps of Pike with the measurement systems of Johnson/Gray in order to provide array-based control of the measurement devices.

Regarding claim 42, Johnson/Gray teaches every limitation of claim 42 except generating an analysis object so that the analysis object can be used in MATLAB. Pike teaches that the graphical user interface can be used to export data to an array-based environment such as MATLAB (see paragraph [0040]; “User 30 may send a list of requests or commands to processor 20 from the GUI 14 to establish a communication channel between the computer 12 and the control instruments 22. The user 30 does so by writing a user program 80, which resides in memory 26 of computer 12. The user program 80 may be associated with the syntax of, for example, any interpreted programming environment. An interpreted programming environment may be any proprietary program that performs mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments, among many others. An example of an interpreted programming environment is MATLAB.RTM. from MathWorks, Inc., of Natick, Mass”). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to provide MATLAB support as taught by Pike to the measurement systems of Johnson/Gray in order to provide MATLAB support for the measurement devices.

Claim 55 recites a system having substantially the same limitations as claim 41.

Therefore, claim 55 is rejected under the same rationale.

Response to Arguments

Applicant has amended claims 30 and 50 to correct minor informalities.

Accordingly, the objections to the claims are withdrawn.

Applicant asserts that Johnson fails to disclose instructions for receiving, from a user, a selection of at most one configuration from the plurality of configurations and instructions for communicating with the hardware device corresponding to the selected configuration using the selected configuration. Examiner respectfully disagrees.

The instant specification does not explicitly define "configuration". As recited in the claims, a "configuration" is defined as "allowing the user to edit at least one property of the hardware object". The broadest reasonable interpretation of "configuration" in light of the specification can be any property setting for the device as shown in Johnson. For example, a channel, a timing, an input range, or a custom scaling setting are all "configurations" falling under the broadest reasonable interpretation in light of the specification because each one of those configurations allow the user "to edit at least one property of the hardware object". Given the recited claim language, a configuration can also be a single property of the hardware object. Examiner does not change the definition of "configuration" within the rejection, but only changes the example used in to reject the limitation. Selecting and changing a single setting from a channel in Johnson also reads on applicants' recited "receiving, from a user, a selection of at most one

Art Unit: 2175

configuration from the plurality of configurations” and “instructions for communicating with the hardware device corresponding to the selected configuration using the selected configuration”.

Applicant cites the Advisory Action filed April 28, 2009 as stating that a configuration is being interpreted as being a “channel” as taught by Johnson. However, the same Advisory Action also states that while a “channel” is an example of a configuration, an individual setting such as “Output Voltage for Logic Low” or “Output Voltage for Logic High” can also be interpreted as individual configurations. The Advisory Action further states, “Examiner can find no teaching in the instant specification that defines a ‘configuration’ as being the complete set of settings required to make a hardware device operate. A ‘configuration’ can be read as a single setting of a hardware device, each hardware device requiring a plurality of configurations to be set in order to operate”. Examiner maintains that a “configuration” as recited in the instant claims can be any single setting or group of settings such as a channel, a timing, an input range, or a custom scaling setting, among others.

When a user selects any single setting in Johnson such as, for example, a custom scaling setting, the processor is “receiving, from a user, a selection of at most one configuration from the plurality of configurations”, as recited in the claim.

Applicant asserts that the items depicted in the Custom Scale dropdown box do not fulfill several of the requirements for a configuration recited in claim 1, such as displaying the plurality of configurations simultaneously, wherein each configuration

Art Unit: 2175

corresponds to a unique hardware object that represents the hardware device.

Examiner respectfully disagrees.

Applicant argues that the configurations in the drop-down menu do not correspond to a unique hardware object that represents the hardware device. Instead, Applicant argues that the menu elements are different scaling functions that may be applied to acquired data. This argument is not found to be persuasive because Johnson that "each of the channels corresponds to a terminal of a corresponding device" (Johnson paragraph [0016]) and that "a scaling expert may be operable to make changes to the run-time specification to specify or implement custom scaling operations for a measurement channel" (Johnson paragraph [0163]). A custom scale is specified for a measurement channel, and each channel corresponds to a device, therefore each custom scale configuration corresponds to a unique hardware object.

Applicant further argues that Johnson is silent with respect to hardware objects, and the custom scale options do not qualify as hardware objects. However, "hardware objects" can be interpreted to mean many different things in the art. Instant specification page 8 states, "The hardware object is representative of a hardware device, is depicted in the graphical interface, and is configured to be interactive with the hardware device and enable communication between the graphical interface and the hardware device". As established above, a custom scale configuration is representative of a hardware device, and is configured to be interactive with the hardware device. As shown in Johnson Figure 19, it is also depicted in the graphical interface. The custom

Art Unit: 2175

scale dropdown box enables communication between the graphical interface and the hardware device.

Applicant asserts that Johnson does not communicate with the hardware device using the selected configuration because the selected scaling function is used after Johnson's program has already received signals from the channel of the hardware device. Examiner agrees with this argument, but it does not overcome Johnson's prior art.

For example, Johnson Figure 18A shows two configurations, Min and Max, allowing the user to select one of the configurations for editing. The user can select and edit one of the two configurations (receiving step) to be communicated to the hardware device (communicating step).

Applicant asserts that neither Johnson nor Gray discloses or suggests providing a graphical interface that provides interaction with an array-based environment. Examiner respectfully disagrees.

An "array-based environment" is not a well-known term in the art. The instant disclosure only describes it on page 24, "Examples of array-based environments can include MATLAB or other interpretive programming environments capable of interfacing with one or more arrays". Johnson teaches that the programming environment may be interpreted (as opposed to compiled) (Johnson paragraph [0185]; *"the program initiation 1210 may involve the user initiating compilation or interpretation of the source program*

Art Unit: 2175

1208"). An "array" as commonly known in the art is a basic data structure that can index a plurality of items. Johnson states that one embodiment of the invention may be a C program, which is well-known in the art as being capable of interfacing with one or more arrays (Johnson paragraph [0185]; *"where the measurement task source program 1208 is a text-based program, e.g., a C program"*). According to the definition of an array-based environment provided in the instant disclosure, it is believed that Johnson can reasonably be interpreted as teaching an array-based environment.

The instant disclosure further discusses capabilities of array-based environments, "Changes that are made to the software objects, hardware objects and analysis objects in the array-based environment 250 are reflected in the graphical interface 200. For example, if the hardware object is disconnected from an associated hardware device from within the array-based environment 250, the graphical interface 200 is updated to indicate that the hardware object is no longer connected to the hardware device. Similarly, if a software object is created in the array-based environment 250, the graphical interface 200 may be updated to include the software object." Gray was cited to disclose these features of the array-based environment, such that if a device is disconnected from the system, the changes will be reflected.

Applicant asserts that neither Johnson nor Gray teach updating the graphical interface when the hardware object or software object are changed in the array-based environment. Examiner respectfully disagrees.

In Johnson, when the measurement device is configured and in use, it monitors conditions at the measurement device and updates the graphical interface to alert users of the changes in the device. For example, Johnson Figures 32A-B shows the graphical interface being updated in response to changes in the measurement device. Therefore the hardware and software objects (as opposed to devices) are changed in the graphical interface.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure:

- Loughlin et al. (US 7,467,372) Device configuration and management development system

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Art Unit: 2175

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Stephen Alvesteffer whose telephone number is (571)270-1295. The examiner can normally be reached on Monday-Friday 9:30AM-6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Bashore can be reached on (571)272-4088. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Stephen Alvesteffer
Examiner
Art Unit 2175

/S. A./
Examiner, Art Unit 2175

/Joshua D Campbell/
Primary Examiner, Art Unit 2178